



CENTRO UNIVERSITÁRIO DE BRASÍLIA -UniCEUB
CURSO DE ENGENHARIA DE COMPUTAÇÃO

DYEGO DINIZ DE SOUZA

CONTROLE DE ILUMINAÇÃO E ENTRADA PRINCIPAL

Orientador: Profa. M. C. Maria Marony Sousa Farias

Brasília
Novembro, 2013
DYEGO DINIZ DE SOUZA

CONTROLE DE ILUMINAÇÃO E ENTRADAS PRINCIPAIS

Trabalho apresentado ao Centro
Universitário de Brasília
(UniCEUB) como pré-requisito
para a obtenção de Certificado
de Conclusão de Curso de
Engenharia de Computação.
Orientador: Profa. M. C. Maria
Marony Sousa Farias

Brasília
Novembro, 2013

DYEGO DINIZ DE SOUZA

CONTROLE DE ILUMINAÇÃO E ENTRADAS PRINCIPAIS

Trabalho apresentado ao Centro
Universitário de Brasília
(UniCEUB) como pré-requisito
para a obtenção de Certificado
de Conclusão de Curso de
Engenharia de Computação.
Orientador: Profa. M. C. Maria
Marony Sousa Farias

Este Trabalho foi julgado adequado para a obtenção do Título de Engenheiro de Computação,
e aprovado em sua forma final pela Faculdade de Tecnologia e Ciências Sociais Aplicadas -
FATECS.

Prof. Maria Marony
Mestre, UniCeub

Banca Examinadora:

Prof. Vera Farini
UniCeub

Prof. Luis Cláudio
UniCeub

Luciano Duque
UniCeub

DEDICATÓRIA

Dedico este trabalho a minha família que me deu tanto apoio durante o curso de graduação, me aconselhando em todas as decisões referentes a este trabalho.

AGRADECIMENTOS

À minha família, por estar sempre ao meu lado incentivando os estudos, me aconselhando para que eu possa evoluir e melhorar cada vez mais em minha vida pessoal, acadêmica e profissional.

Aos meus amigos e colegas que me apoiaram muito para a evolução e conclusão deste projeto.

Aos meus professores que tiveram sempre um tempo para conversar sobre futuro, mercado de trabalho, vida profissional e orientação no projeto final.

E à todos que colaboraram de alguma forma com o desenvolvimento e a realização deste trabalho.

SUMÁRIO

SUMÁRIO.....	VI
CAPÍTULO 1 – INTRODUÇÃO.....	12
1.1 APRESENTAÇÃO DE PROBLEMA.....	12
1.2 OBJETIVOS DO TRABALHO.....	12
1.3 JUSTIFICATIVA E IMPORTÂNCIA DO TRABALHO.....	13
1.4 ESCOPO DO TRABALHO.....	13
1.5 RESULTADOS ESPERADOS.....	14
1.6 ESTRUTURA DO TRABALHO	14
CAPÍTULO 2 – APRESENTAÇÃO DO PROBLEMA	16
2.1 FALHA HUMANA	16
2.2 AGILIDADE EM PROCEDIMENTOS E PROCESSOS	17
2.3 VISÃO GERAL DO PROBLEMA	17
2.4 BENEFÍCIOS DO DISPOSITIVO PROPOSTO E SUAS RESTRIÇÕES	19
CAPÍTULO 3 – REFERENCIAL TEÓRICO	21
3.1 PROTOCOLO TCP/IP	21
3.2 ETHERNET.....	22
3.3 REDES WIRELESS	22
3.4 MICROCONTROLADORES	24
3.5 ARDUINO	24
3.6 SHIELDS ARDUINO.....	25
3.7 INSTALAÇÕES ELÉTRICAS.....	26
3.8 DISJUNTOR TERMOMAGNÉTICO.....	26
3.9 DISJUNTOR DIFERENCIAL RESIDUAL.....	26
3.10 LIGAÇÃO THREE-WAY	26
3.11 NAVEGADORES DA WEB	27
3.12 HTML.....	28
CAPÍTULO 4 – DESCRIÇÃO DE HARDWARE E SOFTWARE.	29
4.1 ARDUINO IDE	29
4.2 ARDUINO UNO	30
4.3 PINAGEM UTILIZADA.....	32
4.4 PLACA COM MÓDULO ETHERNET	32
4.5 PLACA COM MÓDULOS RELÉS	33
4.6 SENSOR DE CORRENTE.....	34
4.7 ROTEADOR WIELESS DI-524	35

4.8	APLICAÇÃO WEB.....	36
CAPÍTULO 5 – DESENHO E IMPLEMENTAÇÃO.....		38
5.1	DESENHO DO PROJETO.....	38
5.2	PLANTA BAIXA E CORTES DO PROTÓTIPO	39
5.3	DESENHO DO PROJETO ELETRÔNICO DO PROTÓTIPO	39
5.4	DESENHO DO PROJETO ELÉTRICO DO PROTÓTIPO.....	40
5.5	ESCRITA DO CÓDIGO FONTE.....	40
5.6	MONTAGEM E TESTES DO PROJETO	42
5.7	MONTAGEM E TESTES DA MAQUETE DO PROTÓTIPO	43
5.8	MONTAGEM E TESTES DA PARTE ELÉTRICA E ELETRÔNICA DO PROTÓTIPO	43
5.9	MONTAGEM E CONFIGURAÇÃO DA TOPOLOGIA DA REDE.....	44
CAPÍTULO 6 – RESULTADOS OBTIDOS		46
6.1	TESTES COM O SISTEMA	46
6.2	TESTES COM A PARTE ELÉTRICA	46
6.3	TESTES COM O CONTROLE.....	47
6.4	ACIONAMENTO DAS LUZES E PORTA PRINCIPAL.....	48
6.5	PROBLEMAS ENCONTRADOS.....	50
CAPÍTULO 7 – CONSIDERAÇÕES FINAIS		52
7.1	CONCLUSÕES	52
7.2	PROPOSTAS PARA TRABALHOS FUTUROS	52
REFERÊNCIAS BIBLIOGRÁFICAS		54
APÊNDICE A – GASTOS DO PROJETO.....		55
APÊNDICE B – CÓDIGO FONTE		56
APÊNDICE C – ESQUEMÁTICO: ARDUINO UNO.....		58

LISTA DE FIGURAS

Figura 1.1 - Visão geral do sistema de controle. Fonte: Autor.....	14
Figura 2.1 - Procedimentos para abrir laboratórios e ligar luzes. Fonte: Autor	18
Figura 2.2 - Procedimentos para fechar laboratórios e desligar luzes. Fonte: Autor	18
Figura 2.3 - Forma otimizada de como a solicitação irá funcionar. Fonte: Autor.....	19
Figura 3.1 - Arquitetura Ethernet. Fonte: Autor	22
Figura 3.2 - Pilha de protocolos TCP/IP com arquitetura Ethernet. Fonte: Autor	22
Figura 3.3 - Arquitetura Wi-Fi. Fonte: Autor	23
Figura 3.4 - Pilha de protocolos TCP/IP com arquitetura Wi-Fi. Fonte: Autor	23
Figura 3.5 - Placa Arduino UNO. Fonte: Autor	25
Figura 3.6 - Ligação Three Way. Fonte: www.eletrica.info	27
Figura 3.7 - Implementação do código processing e HTML.....	28
Figura 4.1 - Arduino IDE. Fonte: Autor	30
Figura 4.2 - Arduino Uno. Fonte: Arduino.cc	31
Figura 4.3 - Shield Ethernet. Fonte: Autor	33
Figura 4.4- Placa com módulos Reles. Fonte: Autor.....	34
Figura 4.5 - Sensor de Corrente. Fonte: Autor	35
Figura 4.6 - Ponto de Acesso. Fonte: Autor	36
Figura 4.7 - Aplicação. Fonte: Autor.....	37
Figura 5.1 - Desenho da planta no Autocad. Fonte: Autor.....	38
Figura 5.2 - Parte do código fonte. Fonte: Autor.....	40
Figura 5.3 - Parte do código fonte. Fonte: Autor.....	41
Figura 5.4 - Parte do código fonte. Fonte: Autor.....	42
Figura 5.5 - Protótipo completo. Fonte: Autor	43
Figura 5.6 - Configuração da criptografia utilizada. Fonte: Autor	44
Figura 5.7- Configuração do filtro de endereço MAC. Fonte: Autor.....	45
Figura 6.1 - Teste de circuito com multímetro. Fonte: Autor.....	47
Figura 6.2 - Shield Ethernet em funcionamento com Arduino Uno. Fonte: Autor	48
Figura 6.3 - Alimentação de 9v para o Arduino. Fonte: Autor	49
Figura 6.4 - Senha de segurança para acesso nos diferentes dispositivos. Fonte: Autor.....	49
Figura 6.5 - Painel de controle em ambos os equipamentos. Fonte: Autor.....	50

LISTA DE TABELAS

Tabela 1 – Funções do arduino IDE. Fonte: Autor	32
Tabela 2 – Principais características Arduino Uno.....	33
Tabela 3 – Pinagem Utilizada para o projeto.....	34

RESUMO

Este projeto apresenta um sistema de controle para luminárias e portas principais controlado de forma remota por qualquer dispositivo que tenha um navegador web, como celulares, notebooks e computadores. Para controle do protótipo é utilizado a plataforma micro controlada Arduino UNO, cujo microcontrolador é o ATmega328 trabalhando em conjunto com um Shield Ethernet e uma placa com módulos relés. Através dos dispositivos que controlam o protótipo podemos ligar e desligar as luminárias de seus cômodos e abrir sua porta principal. O protótipo, ao receber algum comando, envia um pulso elétrico ao módulo relé acionando a lâmpada de algum cômodo ou a porta principal. A linguagem de marcação utilizada para criar o painel principal de controle foi o HTML e a linguagem de programação utilizada para o desenvolvimento dos controles foi a nativa do Arduino que implementa processing e C.

Palavras-chave: Automação, Controle Remoto, Arduino, Ethernet, ATmega328.

ABSTRACT

This project presents a control system for main doors and fixtures controlled remotely by any device that has a web browser, such as mobile phones, laptops and computers. To control the prototype is used to Arduino UNO controlled micro-platform, which is the microcontroller ATmega328 working in conjunction with an Ethernet and a shield plate modules relays. Through the devices that control the prototype can turn on and off the fixtures of their rooms and open your front door. The prototype, upon receiving a command, sends an electrical pulse to the module relay triggering the the lamp any room or door. The markup language used to create the master control panel was the HTML and programming language used to develop the controls was the native Arduino that implements processing and C.

Keywords: Automation, Remote Control, Arduino, Ethernet, ATmega328.

CAPÍTULO 1 – INTRODUÇÃO

1.1 APRESENTAÇÃO DE PROBLEMA

Existem processos e procedimentos que não somente podem ser automatizados como devem ser automatizados. Imagine os processos e procedimentos que estão envolvidos na seguinte tarefa: Manter salas com suas luzes ligadas ou desligadas e abrir suas portas.

Boa parte da motivação deste trabalho se resume em uma pequena história. Todos os dias, o monitor ou professor de Sistemas de Comunicação do UNICEUB, tem de ter acesso aos laboratórios 5000, 5001 e 5002. Então, assim que chegar a universidade, a pessoa deve se deslocar até a secretaria do bloco 07 e fazer a solicitação a alguma atendente que realiza o pedido para que as portas dos laboratórios citados sejam abertas. A atendente, por sua vez, solicita ao departamento responsável pelas chaves do bloco 05 que, segundo as secretárias, é o departamento de informática. O departamento, por sua vez, cria outra solicitação para que o funcionário responsável fosse até o local abrir a porta ou portas selecionadas dos laboratórios desejados. Então o funcionário identifica este chamado, vai até outra sala de chaves fazer a busca da chave dos laboratórios, pega a chave, se desloca até onde a pessoa que iniciou todo o processo está esperando e, por fim, abre a porta e liga as luzes dos laboratórios.

Todo esse conjunto de processos e procedimentos leva de 10 a 20 minutos, sem contar algumas vezes que, por alguma falha de comunicação em todo esse processo, o funcionário não chega a atender o chamado de abrir o laboratório e a solicitação tinha que ser feita mais uma vez.

1.2 OBJETIVOS DO TRABALHO

Objetivo Geral – Desenvolver um sistema que implementa automação residencial/comercial.

Objetivos Específicos – Criar um protótipo que simule e utilize as mesmas tecnologias que seriam aplicadas em um ambiente real, de forma a provar que é possível automatizar todos os processos e procedimentos citados no item 1.1 os deixando muito mais ágeis.

1.3 JUSTIFICATIVA E IMPORTÂNCIA DO TRABALHO

Gastar tempo e dinheiro em uma solução que automatize processos e procedimentos que traga agilidade ao negócio, diminua os riscos de falhas, estresse de funcionários e alunos, economia com gastos mensais entre outros benefícios é algo que tinha que ser feito e pensado por todo aluno de engenharia.

Este projeto busca uma solução que seja estável, segura e ágil. Os microcontroladores tem a capacidade de trabalhar 24 horas por dia, 7 dias da semana, não sentem fome, não faltam trabalho, não tiram férias nem as mais diversas licenças de todos os tipos. Por isso suas funcionalidades devem ser aproveitadas com ideias similares a deste projeto.

1.4 ESCOPO DO TRABALHO

Todo o projeto será realizado em um protótipo, com o objetivo de minimizar os custos do projeto otimizando sua apresentação para a banca examinadora. Este protótipo terá sua rede elétrica de corrente alternada, para que sua apresentação fique mais próxima de um ambiente real.

Terá o controle de portas e iluminação através de qualquer dispositivo que tenha acesso a um navegador de internet. Este dispositivo, seja ele um tablet, celular, smart TV, notebook ou computador pessoal terá acesso a um painel de controle que irá controlar as lâmpadas e porta principal.

Esta interface web irá acessar um o programa que irá estar em um Arduino uno com o shield Ethernet. Terei também um ponto de acesso wireless ligado pela interface ethernet do shield ao Arduino, com isso terei acesso ao sistema por qualquer dispositivo que tenho uma placa de rede wireless.

O Arduino irá receber estas informações de controle e, através das portas digitais, mandar comandos para relés, que controlam os circuitos elétricos, como porta e lâmpadas.

Com isso, consigo automatizar tanto um protótipo simples quanto um bloco inteiro do UNICEUB, como por exemplo o bloco 5. A figura 1.1 mostra o funcionamento da comunicação do sistema e suas diferentes arquiteturas.

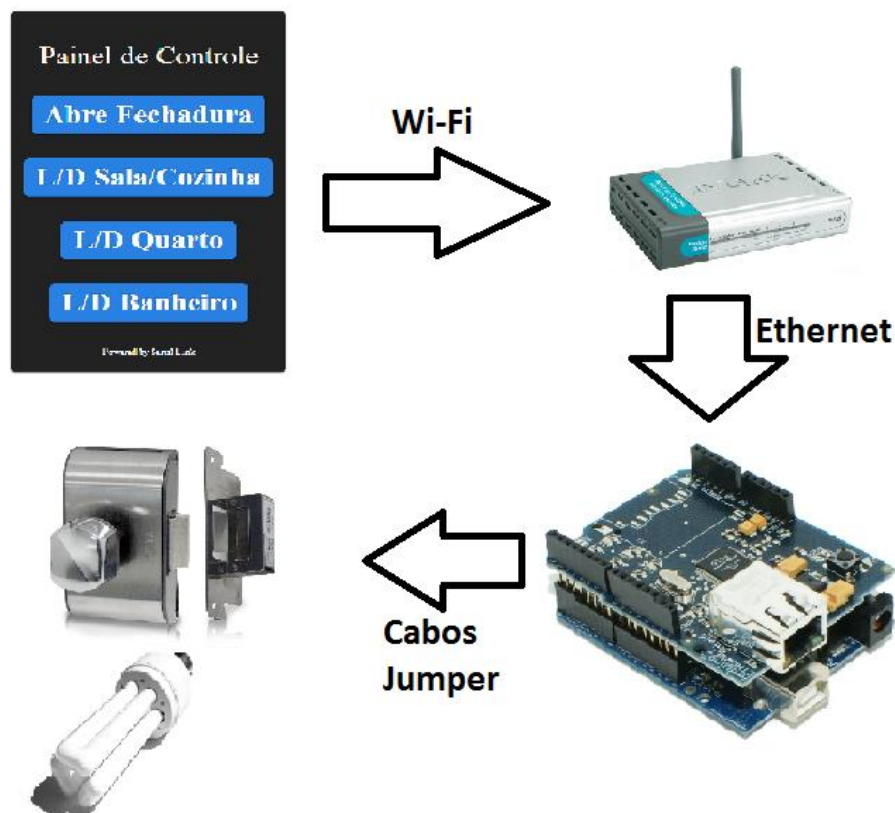


Figura 1.1 - Visão geral do sistema de controle. Fonte: Autor

1.5 RESULTADOS ESPERADOS

Automatizar todos os processos e procedimentos que são realizados e aplicados para ligar as luzes do laboratório e abrir suas portas principais, com isso diminuindo a energia humana, física e mental, o tempo de término do processo e falhas humanas utilizando uma aplicação que seja estável, flexível, segura e com uma comunicação entre os componentes elétricos e eletrônicos bastante eficiente.

1.6 ESTRUTURA DO TRABALHO

Este documento é dividido em 7 capítulos. O primeiro, INTRODUÇÃO, trata da apresentação do problema, dos objetivos do trabalho, justificativa e importância do trabalho, apresenta o escopo do trabalho, os resultados esperados e sua estrutura.

O segundo capítulo, APRESENTAÇÃO DO PROBLEMA, trata a respeito as duas motivações principais que levaram ao desenvolvimento deste projeto, também apresentando os benefícios do dispositivo proposto e suas restrições.

O terceiro capítulo, REFERENCIAL TEÓRICO, aborda todo o embasamento teórico que foi utilizado no projeto para melhor compreensão de desenvolvimento do mesmo, como redes de computadores, microcontroladores, instalações elétricas e programação web.

O quarto capítulo, DESCRIÇÃO DE HARDWARE E SOFTWARE, trás ao leitor detalhes técnicos do que faz parte da parte física e lógica do projeto dando uma maior ênfase ao que será utilizado no projeto.

O quinto capítulo, DESENHO E IMPLEMENTAÇÃO, mostra toda a lógica utilizada para realização de planejamento, desenho e montagem do projeto. A estratégia utilizada neste projeto vem, em detalhes neste capítulo.

O sexto capítulo, RESULTADOS OBTIDOS, contem uma bateria de testes realizados no protótipo construído com o objetivo de ver o comportamento de toas as funcionalidades propostas. Também temos neste capítulo os problemas encontrados no projeto.

O sétimo capítulo, CONSIDERAÇÕES FINAIS, mostra as contribuições do projeto e sugestões para trabalhos futuros.

CAPÍTULO 2 – APRESENTAÇÃO DO PROBLEMA

Este capítulo procura detalhar as motivações para a construção deste projeto, previamente descritas na seção – Apresentação do Problema – do capítulo anterior.

2.1 FALHA HUMANA

Com a explicação do ambiente no capítulo 1 podemos ver a quantidade de pessoas que estão envolvidas no procedimento de abrir a porta e ligar as luzes de apenas um laboratório. Quanto mais pessoas estiverem envolvidas em procedimentos desta forma, a probabilidade de algo dar errado é muito grande.

Por exemplo, a etapa em que o atendente do bloco 07 liga para o departamento de informática, que por sua vez repassa este atendimento para o funcionário das chaves e este funcionário, por alguma falta de comunicação, não recebe este chamado de abertura de laboratórios, neste caso temos a ocorrência de um atraso muito grande em todo o processo. Tanto o atendente que está no bloco 07 quanto o professor ou aluno monitor, que está no aguardo da abertura dos laboratórios iram demorar em identificar esse erro e realizar uma nova solicitação para que o laboratório seja aberto. O mesmo se repete para o processo inverso, ou seja, para desligar as luzes e fechar a porta dos laboratórios. Esse problema pode atrasar algum laboratório planejado por algum professor e, dessa forma, afetar diretamente os alunos de laboratório.

Outro problema, que não afeta diretamente o aluno, mas também pode ocorrer por falha humana é o esquecimento, por parte do funcionário das chaves, de desligar as luzes das salas. Vamos pegar o bloco 05 como exemplo que possui várias salas, o funcionário responsável terá de ir em todas as salas para verificar se a luz está acesa, e em caso positivo, ir desliga-las tudo de forma manual. Imagine esse processo se repetindo pelos outros blocos do Centro acadêmico qual a probabilidade dos funcionários responsáveis pelas chaves esquecerem de alguma sala ou setor específico. Então temos um ambiente com muito trabalho humano e com grandes chances de Falha Humana.

2.2 AGILIDADE EM PROCEDIMENTOS E PROCESSOS

Para a atividade do professor ou aluno monitor fazerem uma solicitação de acesso aos laboratórios e, de fato, alcançarem seu objetivo, é possível automatizar todos os processos e obter um maior controle sobre o sistema.

Processo é uma sequência de atividades predefinidas para que um objetivo maior seja alcançado, este processo possui um início e um fim. Então abrir portas dos laboratórios 5000, 50001 e 5002 passa por um processo que começa pelo aluno monitor realizando a solicitação de abertura de laboratório e tem seu fim na finalização do chamado, registrando o que foi feito, após o funcionário ter aberto a porta dos laboratórios.

Procedimento são os métodos ou metodologias para que esse processo seja feito, existem documentos da própria universidade trazendo informações o que deve ser feito, como deve ser feito, quem deve ser solicitado para tal atividade.

Todo esse conjunto pode ser otimizado através de novos procedimentos e processo utilizando recursos de automação.

2.3 VISÃO GERAL DO PROBLEMA

Para ter uma visão mais abrangente do problema, nas duas figuras a seguir, figuras 2.1 e 2.2, trago dois casos muito comuns. O primeiro deles é a solicitação de abertura dos laboratórios e no segundo temos a solicitação de fechamento dos laboratórios.

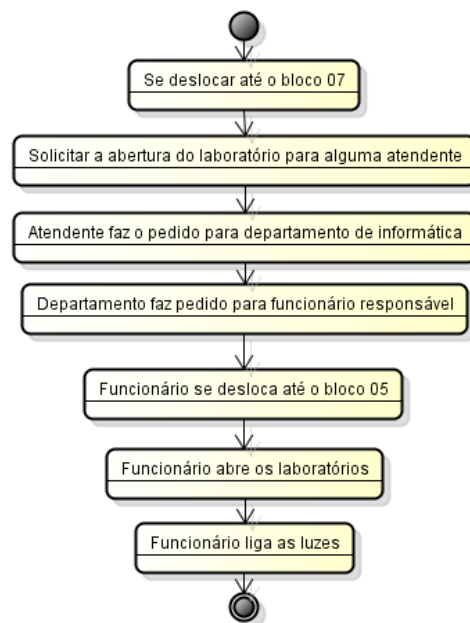


Figura 2.1 - Procedimentos para abrir laboratórios e ligar luzes. Fonte: Autor

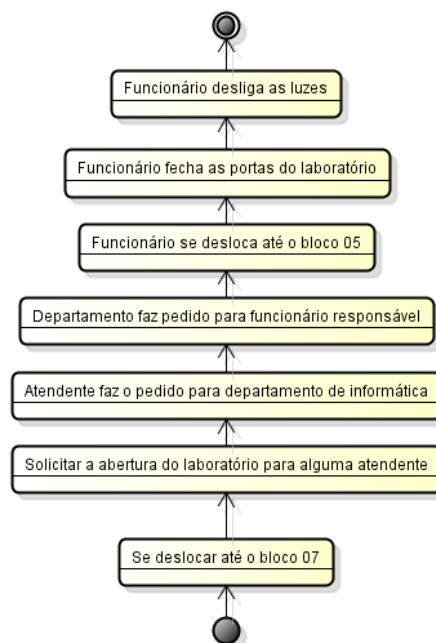


Figura 2.2 Procedimentos para fechar laboratórios e desligar luzes. Fonte: Autor

Este processo de 7 fases dura em média de 10 a 20 minutos. Então o aluno passa esse tempo esperando o funcionário das chaves chegar, abrir a porta e ligar as luzes. Em entrevista com os funcionários, somente aqueles que realizam essa chamada de abertura dos laboratórios, da secretaria do bloco 07 foi informado que há apenas um funcionário com as

chaves dos laboratórios. Então, por dependermos deste funcionário, podemos ter grandes atrasos ou problemas caso aconteça algo com o mesmo.

Outro problema é a probabilidade de ocorrer uma falha na comunicação entre os departamentos ou entre funcionários. Neste caso o aluno iria ficar esperando por mais tempo até que ele mesmo notasse que ocorreu algo de errado, fosse ao bloco 07 verificar e perceber que terá de fazer outro pedido, iniciar todo o processo, para que os laboratórios sejam abertos.

2.4 BENEFÍCIOS DO DISPOSITIVO PROPOSTO E SUAS RESTRIÇÕES

O sistema proposto tem um foco maior em dois benefícios, evitar erros causados por funcionários e agilizar processos para abrir portas, ligar e desligar luzes de forma automatizada. Então os benefícios se resumem em otimizar os dois conjuntos de processos e procedimentos citados nos itens 2.3, figuras 2.1 e 2.2. A figura 2.3 apresenta a nova ideia após a implementação deste sistema.

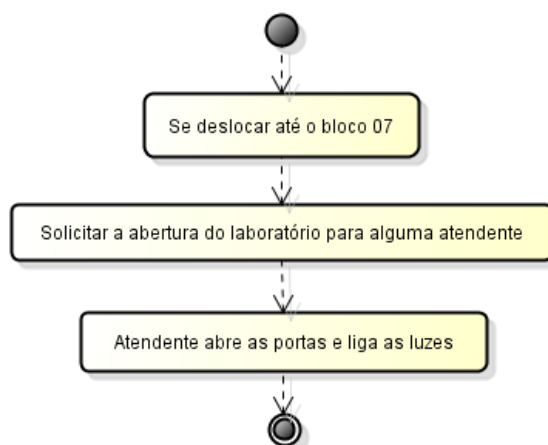


Figura 2.3 - Forma otimizada de como a solicitação irá funcionar. Fonte: Autor

Utilizar este sistema iria reduzir grande parte do tempo que todo esse processo leva para ser realizado, ou seja, teremos um tempo de espera muito menor do monitor na porta do laboratório, desta forma, aumentando a comodidade dos funcionários, alunos monitores e alunos não monitores, todos envolvidos.

O sistema facilita o modo de funcionários de um determinado local abrir as portas e ligarem as luzes de forma automática, através de qualquer dispositivo que tenha acesso ao

navegador web, desta forma tornando todo o processo mais ágil e fácil de ser gerenciado e também reduz o índice de falhas humanas.

Neste projeto não irei abordar a verificação do estado: aberta, fechada, trancada ou destrancada, das portas principais. Com relação as portas, o sistema irá apenas abri-las. Também não será abordado a verificação dos estados das luminárias, ou seja, se estão ligadas ou desligadas.

CAPÍTULO 3 – REFERENCIAL TEÓRICO

Para um melhor entendimento e desenvolvimento deste trabalho são necessários alguns conceitos que se relacionam diretamente ao projeto. Esses conceitos serão apresentados com o foco no trabalho que será realizado.

3.1 PROTOCOLO TCP/IP

Segundo TANEBAUM (2009), “a pilha de protocolos TCP/IP é atualmente a mais utilizada em redes de computadores. Isso por causa de dois fatores. O primeiro deles é a popularização da Internet, já que o protocolo TCP/IP foi criado com fins para ser utilizado na Internet. O segundo e importante fator é que este é um protocolo roteável, isto é, permite o uso de dispositivos chamados roteadores que irão definir o melhor caminho a ser usado pelos pacotes de dados enviados e recebidos dos transmissores a seus receptores. Este recurso é essencial para redes maiores que uma rede local.

Outro fato muito importante que acabou tornando o protocolo TCP/IP muito popular de uma forma bastante rápida é que ele possui arquitetura aberta e qualquer fabricante de dispositivos podem adotar a sua própria versão do TCP/IP em seu sistema operacional, sem a necessidade de pagamento de royalties a ninguém.

Para entender melhor a explicação do protocolo IEEE 802.11 temos 3 tipos de arquiteturas destas redes: Ad-hoc, BSS(Basic Service Set) e ESS(Extended Service Set). Neste projeto, para fazer a comunicação entre os controladores do sistema e o arduíno, será abordado apenas a arquitetura ESS.

Neste modo de operação, BSS(Basic Service Set), a rede sem fio é gerenciada por um periférico chamado ponto de acesso(frequentemente abreviado AP, Acess Point). Então os controladores acessam a rede principal, cabeada, através do ponto de acesso, quando isso ocorre, o nome do modo de operação passa a ser IBSS(Infrastructure Basic Service Set). Neste modo de operação a rede sem fio recebe um nome, chamdo SSID (Service Set ID), que é configurado pelo gestor da rede. Em um computador ou celular com placa de rede sem fio pode ser acessado selecionando a rede a qual queremos nos conectar: o nome da rede que aparece na lista de redes disponíveis é o SSID da rede.”

3.2 ETHERNET

A arquitetura Ethernet é mais utilizada em redes locais, operando nas camadas um e dois do modelo OSI definindo, portanto, a parte física da rede local, segue figura abaixo. Esta arquitetura está atualmente disponível em quatro velocidades máximas de transmissão: 10Mbps (Ethernet padrão), 100Mbps (Fast Ethernet), 1 Gbps (Gigabit Ethernet) e 10 Gbps (10G Ethernet).

Então podemos concluir que o papel do Ethernet é pegar os dados entregues pelos protocolos de alto nível e inseri-los dentro de quadros que serão enviados através da rede. O Ethernet define também como fisicamente esses dados serão transmitidos, como por exemplo o formato do sinal, a figura 3.1 ilustra a modificação da arquitetura Ethernet.

Camada	Nome	Ethernet
2	Link de Dados	CONTROLE DO LINK LÓGICO (LLC) - IEEE 802.2 CONTROLE DE ACESSO AO MEIO (MAC) - IEEE 802.3
1	Física	Física

Figura 3.1 - Arquitetura Ethernet. Fonte: Autor

Seu cenário mais comum é o modelo real apresentado na Figura abaixo, onde a camada de Interface com a Rede do protocolo TCP/IP é o padrão Ethernet.

Camada	Nome	
5	Aplicação	TCP/IP
4	Transporte	
3	Rede	
2.1	CONTROLE DO LINK LÓGICO (LLC) - IEEE 802.2	ETHERNET
2.2	CONTROLE DE ACESSO AO MEIO (MAC) - IEEE 802.3	
1	Física	

Figura 3.2 - Pilha de protocolos TCP/IP com arquitetura Ethernet. Fonte: Autor

3.3 REDES WIRELESS

Existem várias tecnologias para se montar uma rede sem fio, sendo o padrão IEEE 802.11 o mais popular. Este padrão é também conhecido como Wi-Fi.

O padrão IEEE 802.11 é usado para a montagem de redes locais sem fio, usando transmissão por ondas de rádio. A taxa de transferência e o alcance dependem do padrão usado na camada física da rede (IEEE 802.11b, IEEE 802.11g etc) do ambiente e do tipo de antena usado.

Este padrão opera nas camadas um e duas do modelo OSI e, portanto, responsável por pegar os pacotes de dados passados pelo protocolo de mais alto nível usado, dividi-los em quadros e transmiti-los via ondas de rádio. Na figura 3.3 podemos ver a arquitetura do padrão IEEE802.11.

Camada	Nome	Wi-Fi (IEEE 802.11)
2	Link de Dados	CONTROLE DO LINK LÓGICO (LLC) - IEEE 802.2 CONTROLE DE ACESSO AO MEIO (MAC) - IEEE 802.11
1	Física	Física - IEEE 802.11b, IEEE 802.11g, etc.

Figura 3.3 - Arquitetura Wi-Fi. Fonte: Autor

Uma representação mais real é apresentada na figura 3.4, onde mostro o modelo em camadas de uma rede Wi-fi usando a pilha de protocolos TCP/IP, que é o cenário mais comum visto em projetos.

Camada	Nome	
5	Aplicação	TCP/IP
4	Transporte	
3	Rede	
2.1	CONTROLE DO LINK LÓGICO (LLC) - IEEE 802.2	Wi-Fi
2.2	CONTROLE DE ACESSO AO MEIO (MAC) - IEEE 802.11	
1	Física - IEEE 802.11b, IEEE 802.11g, etc.	

Figura 3.4 - Pilha de protocolos TCP/IP com arquitetura Wi-Fi. Fonte: Autor

As três camadas da arquitetura Wi-Fi e Ethernet possuem as seguintes funções:

*Controle do Link Lógico (LLC, IEEE 802.2): Aqui temos informações do protocolo de alto nível, ou seja o que está operando na camada de REDE, que entregou o pacote de dados a ser transmitido. Com isso, a máquina receptora tem como saber para qual protocolo de alto nível ela deve entregar os dados de um quadro que ela acabou de receber.

*Controle de Acesso ao Meio (MAC, IEEE 802.11): Aqui temos uma montagem do quadro de dados que será transmitido pela camada física, incluindo cabeçalhos próprios dessa

camada aos dados recebidos da camada de Controle do Link Lógico. Também temos nessa camada uma verificação se o meio, no caso o ar, está livre para a transmissão de dados.

*Física: Aqui temos a transformação dos bits, isto é, sinais elétricos, enviados pela camada de controle de acesso ao meio em ondas de rádio. Padrões diferentes operam nesta camada, tais como o IEEE 802.11a, IEEE 802.11b, IEEE 802.11g e IEEE 802.11n, que definem a frequência do espectro que será usada e a técnica de transmissão.

3.4 MICROCONTROLADORES

Segundo Gimenez (2005), “o microcontrolador é um dispositivo semicondutor em forma de circuito integrado, que integra as partes básicas de um microcomputador – microprocessador, memórias não-voláteis e voláteis e portas de entrada e saída”.

Utilizamos microcontroladores para tarefas que não exigem grande capacidade de processamento, mas um controle sobre algum sistema. Portanto são de grande utilidade na área de automação.

3.5 ARDUINO

Arduino é uma plataforma microcontrolada, um pequeno computador que podemos programar para processar entradas e saídas entre o dispositivo e os componentes externos conectados a ele. O Arduino é o que chamamos de plataforma de computação embarcada, ou seja, um sistema que pode interagir com o ambiente físico por meio de hardware e software. Por exemplo, o Arduino poderia ser utilizado para acionar um motor ou um led por um determinado tempo, digamos, um minuto, depois que um botão fosse pressionado por alguém. O Arduino aguardaria até que o botão fosse pressionado; uma vez pressionado o botão, ele acionaria o motor e iniciaria a contagem. Depois de contados os 60 segundos, desligaria o motor e iria ficar no aguardo de outro acionamento do botão. Poderíamos utilizar essa configuração para controlar a esteira de um robô, por exemplo. Segue adiante, na figura 3.5, uma imagem da plataforma.

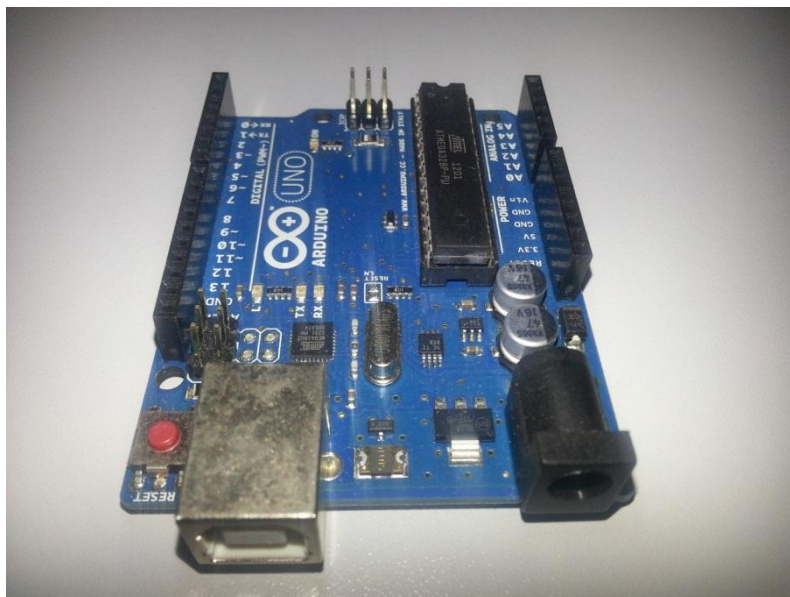


Figura 3.5 - Placa Arduino UNO. Fonte: Autor

Podemos estender este conceito modificando o acionamento do motor com um botão para o acionamento por um sensor de temperatura, para o motor ligar quando o sensor identificasse uma temperatura pré-determinada. Esses são exemplos simples de como utilizar um Arduino.

O arduino pode ser utilizado de forma independente com objetos iterativos, pode ser conectado ao computador ou celular para atuar, enviar e recuperar dados sobre ele. Em outras palavras, ele pode receber um conjunto de dados de sensores ou outros dispositivos, trata-los e apresentar como um gráfico para algum site ou programa no computador.

3.6 SHIELDS ARDUINO

Também podemos estender as funcionalidades do Arduino através dos Shields (escudos), que são placas de circuito contendo outros dispositivos, como por exemplo, módulos GSM, displays de LCD, receptores wireless etc... que podemos conectar ao Arduino. Os Shields também estendem os pinos até o topo de suas próprias placas de circuito, para que possamos ter acesso ao Arduino. Não é obrigatório o uso de um shield para o Arduino, apenas quando temos a necessidade de estender suas funcionalidades. Como por exemplo implementar a funcionalidade de ligar e enviar mensagens para que o Arduino possa interpretá-las e fazer ações de acordo com os comandos enviados ou números discados na

ligação. Desta forma podemos controlar através do celular por causa da funcionalidade adicional que o Shield GSM do arduíno proporciona.

3.7 INSTALAÇÕES ELÉTRICAS

O projeto funciona em um protótipo, cuja seus interruptores e instalações são feitas em corrente alternada, estabilizada e com tensão de 110V. Os Disjuntores e Interruptores são os dispositivos centrais no controle de toda a parte elétrica do projeto.

3.8 DISJUNTOR TERMOMAGNÉTICO

São dispositivos que permitam manobrar o circuito elétrico, Operando-o como um interruptor, liga e desliga somente o circuito necessário para uma eventual manutenção. Estes equipamentos também oferecem proteção aos condutores que estão ligados ao circuito, quando temos uma sobrecorrente provocada por um curto-circuito ou sobrecarga o DTM o desliga de forma automática.

Para entendermos melhor o funcionamento dos disjuntores termomagnéticos podemos compara-los as chaves fusíveis com uma grande diferença a ser comentada. Nas chaves fusíveis o fusível se queima necessitando ser substituído por outro. Já no caso do DTM, ele se desliga necessitando apenas religa-lo para coloca-lo em operação.

3.9 DISJUNTOR DIFERENCIAL RESIDUAL

É um dispositivo que possui um disjuntor termomagnético acoplado com outro dispositivo: o diferencial residual. Desta forma este dispositivo conjuga duas funções: A do disjuntor termomagnético protegendo os condutores do circuito contra curto-circuito e sobrecarga e a do dispositivo diferencial residual protegendo as pessoas contra choques elétricos provocados por contatos diretos e indiretos.

3.10 LIGAÇÃO THREE-WAY

Utiliza-se um conjunto de interruptores em paralelo para a necessidade de controlar uma ou várias lâmpadas situadas no mesmo ponto, de mais um local diferente, esta ligação é

conhecida também como ligação Three Way. Esse Interruptor além de proporcionar um maior conforto para o usuário, aumenta os aspectos quanto a segurança, devido ao comando da iluminação estar em mais de um ponto. Essa ligação, entre um interruptor o relé, pode ser vista na figura 3.6.

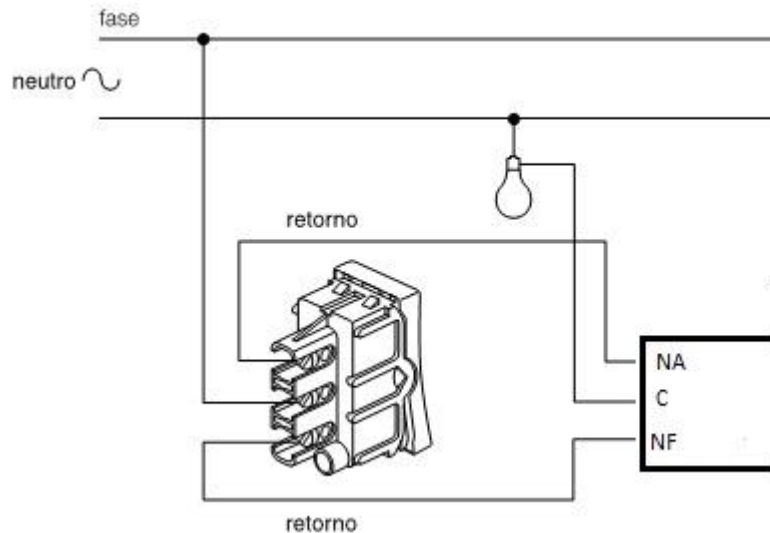


Figura 3.6 - Ligação Three Way. Fonte: www.eletrica.info

3.11 NAVEGADORES DA WEB

Segundo Lemay (2002), “um navegador da Web é um programa que você usa para visualizar páginas e navegar na World Wide Web. Os navegadores da Web são chamados, algumas vezes, de clientes da web ou outros nomes da moda (ferramentas de navegação na Internet), mas navegador da web é o termo mais comumente utilizado.”

A página do painel de controle para controlar o protótipo automatizado irá estar em um navegador web. De lá podemos ligar, desligar as luminárias e fazer a abertura da porta principal.

3.12 HTML

Segundo Lemay (2002), “O HTML é uma linguagem para a descrição da estrutura de um documento e não sua verdadeira apresentação. A ideia aqui é que a maior parte dos documentos possua elementos em comum – por exemplo, títulos, parágrafos ou listas”.

Todo o texto, e sua organização, no painel de controle da aplicação no navegador web é estruturado pelo HTML. Porém também está contido na parte de programação a linguagem de programação *processing*, muito similar a linguagem de programação C. Todo o HTML é gerado pela segunda linguagem citada. A figura abaixo mostra como é feita a integração do código HTML na linguagem de programação *processing*.

```
sensorQuarto = analogRead(A5);
client.print("<p>Status Quarto: "); client.print(sensorQuarto); client.println("</p>");
sensorBanheiro = analogRead(A4);
client.print("<p>Status Banheiro: "); client.print(sensorBanheiro); client.println("</p>");
sensorSalaCozinha = analogRead(A3);
client.print("<p>Status Sala/Cozinha: "); client.print(sensorSalaCozinha); client.println("</p>");

//GERA TODO O HTML dos botões
client.println("<p><a href=\"192.168.1.177/ativaporta\">ABRE PORTA<a></p>");
client.println("<p><a href=\"192.168.1.177/ativasalacozinha\">L/D SALA/COZINHA<a></p>");
client.println("<p><a href=\"192.168.1.177/ativaquarto\">L/D QUARTO<a></p>");
client.println("<p><a href=\"192.168.1.177/ativabanheiro\">L/D BANHEIRO<a></p>");
```

Figura 3.7 - Implementação do código *processing* e HTML

CAPÍTULO 4 – DESCRIÇÃO DE HARDWARE E SOFTWARE.

Para entender melhor como o Hardware e o Software trabalham, separei o projeto em sistema de controle e controladores.

O sistema de controle inclui o WebServer que recebe as requisições (comandos) dos clientes para realizar o controle sobre o sistema controlado, no caso nosso protótipo de residência automatizada. Outra função muito importante do WebServer é apresentar o estado das luminárias para os clientes.

Os controladores são os clientes que fazem as requisições e verificam os estados das luminárias. Este cliente pode ser qualquer equipamento que tenha um navegador web, como por exemplo, o Mozilla Firefox ou Google Chrome.

4.1 ARDUINO IDE

Para programar o Arduino, fazer com que ele controle equipamentos da forma desejada, é utilizado o IDE do Artuíno, um software livre no qual é escrito o código na linguagem que o Arduino compreende, esta linguagem é baseada na linguagem de programação C. Esta interface de desenvolvimento permite a escrita de um programa de computador através de um conjunto de instruções passo a passo e logo após e feito o upload para a plataforma Arduino. Após este procedimento, o Arduino irá executar as instruções descritas, interagindo com o sistema conectado a ele. No Arduino estes exemplos de programas são conhecidos como sketches, rascunhos ou esboço. A figura 4.1, traz a interface da interface de desenvolvimento do Arduino numerada pelo autor para uma melhor descrição na próxima tabela.

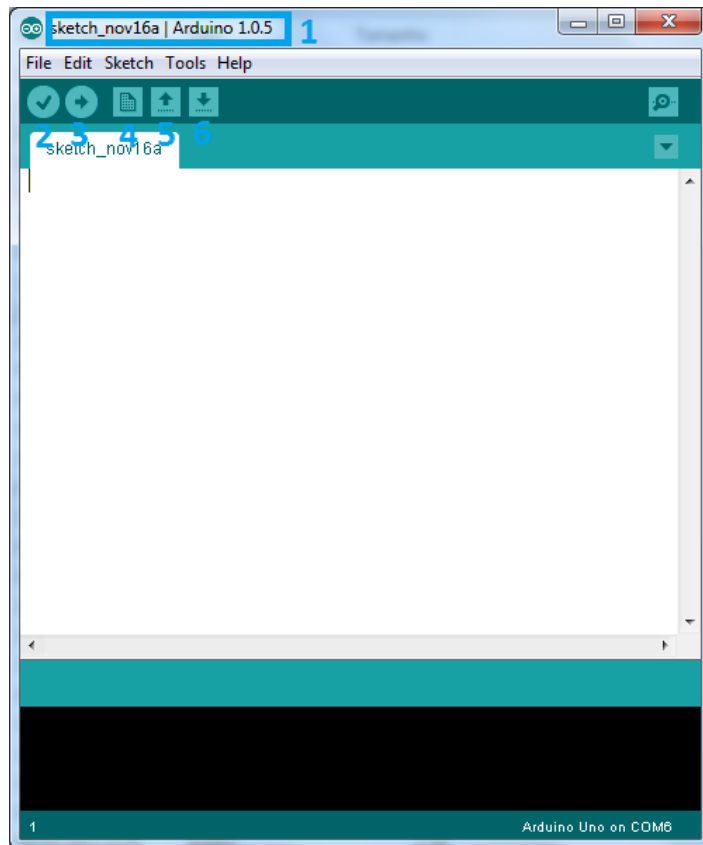


Figura 4.1 - Arduino IDE. Fonte: Autor

Número	Nome	Descrição
1	Título Versão	Composto do nome que será dado ao código e da versão da interface de desenvolvimento.
2	Verify	Compila o código fonte
3	Upload	Realiza a compilação do código fonte e o envia para a placa Arduino UNO
4	New	Abre uma nova interface para desenvolvimento
5	Open	Abre uma interface que estava em desenvolvimento
6	Save	Salva o código que está em desenvolvimento

Tabela 1 – Funções do arduino IDE. Fonte: Autor

4.2 ARDUINO UNO

O modelo do Arduino utilizado neste trabalho é o Arduino UNO. Basicamente este componente é uma plataforma microcontrolada baseado no chip ATmega328. Possui 14 pinos

de entrada/saída digitais e 6 pinos de entradas analógicos, uma conexão USB, um conector de alimentação e um botão de reset.

A versão Uno é diferente de todas as placas anteriores, como o duemilanove por exemplo, pois não utiliza o chip FTDI USB-to-Serial. Em vez disso, ele apresenta o Atmega16U2 programado com um conversor USB-to-Serial.



Figura 4.2 - Arduino Uno. Fonte: Arduino.cc

As principais características desta versão são apresentadas na tabela abaixo:

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	5-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (Atmega 328)
SRAM	2 KB (ATmega 328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 Mhz

Tabela 2 – Principais características Arduino Uno.

4.3 PINAGEM UTILIZADA

A tabela abaixo detalha a pinagem utilizada descrevendo o pino e sua funcionalidade específica neste projeto.

Pino	Utilização
2 Digital	Controle do Rele K1 (Fecho Elétrico)
4 Digital	Controle do Rele K2 (Banheiro)
6 Digital	Controle do Rele K3 (Sala)
8 Digital	Controle do Rele K4 (Quarto)
1 Analógico	Verifica Sensor de corrente S3 (Banheiro)
3 Analógico	Verifica Sensor de corrente S2 (Quarto)
5 Analógico	Verifica Sensor de corrente S1 (Sala/Cozinha)
5V	Barramento 5v
Gnd	Barramento Gnd

Tabela 3 – Pinagem Utilizada para o projeto.

4.4 PLACA COM MÓDULO ETHERNET

O shield Ethernet possui um conector RJ-45 que permite a ligação do Arduino a uma rede de computadores.

A ultima versão desta mesma placa também inclui um slot para cartão micro-SD, que pode ser conectado com o uso da biblioteca SD, mas esta funcionalidade não será aplicada neste trabalho. Abaixo temos a imagem do Shield Ethernet.

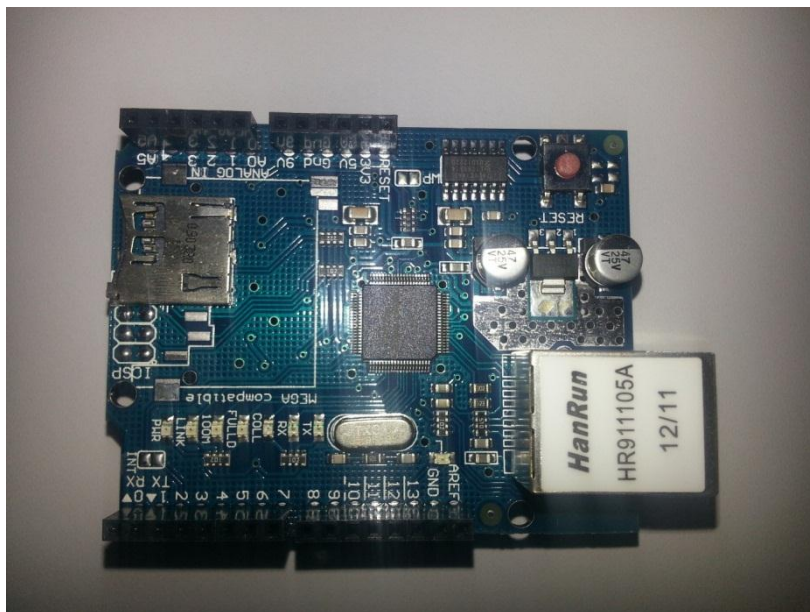


Figura 4.3 - Shield Ethernet. Fonte: Autor

O acesso ao Shield Ethernet é feito por qualquer dispositivo que tenha um navegador de internet instalado para acessar a página web com o ip configurado no arduino. Esta página se atualiza a cada cinco segundos para que as configurações feitas no Shield Ethernet sejam visualizadas.

4.5 PLACA COM MÓDULOS RELÉS

O relé é um dispositivo eletromecânico, com várias aplicações em comutação de contatos elétricos, na grande maioria das vezes, servindo para ligar ou desligar dispositivos. A topologia normal de se encontrar na aplicação de relés é encontra-los ligados em dois circuitos, como será feito neste projeto. A comutação, nestes relés, é realizada alimentando-se a bobina do mesmo pelo circuito de controle. Quando uma determinada corrente originada no circuito de controle passa pela bobina, um campo eletromagnético é gerado, acionando o relé e possibilitando o funcionamento de um segundo circuito, o circuito controlado. Então a aplicabilidade do relé neste projeto é utilizar baixas correntes para o comando no circuito de controle para controlar um segundo circuito.

Neste projeto será utilizado um módulo com 4 relés identificados por K1, K2, K3 e K4 respectivamente. Cada relé possui 4 entradas: Comum, Normalmente Aberto,

Normalmente Fechado, e Neutro. Alguns módulos reles não vêm com estas saídas para o neutro do circuito elétrico. Este, por ser muito compacto, vem com esta saída para diminuir as interferências geradas pelo circuito. Abaixo temos uma imagem do módulo com 4 relés.



Figura 4.4 - Placa com módulos Reles. Fonte: Autor

4.6 SENSOR DE CORRENTE

O sensor de corrente possui o chip: ACS712ELC-5^a, tem alimentação de 5v, com indicação on-board por led e seu módulo pode medir até 5 A positivo ou negativo.

Seu funcionamento é muito simples, este componente está ligado ao retorno de cada lâmpada para verificar se há ou não passagem de corrente pelo fio. Caso ocorra passagem de corrente pelo fio ele envia essas informações de forma analógica para o Arduino. Segue abaixo, na figura 4.5, uma foto do sensor de corrente utilizado.

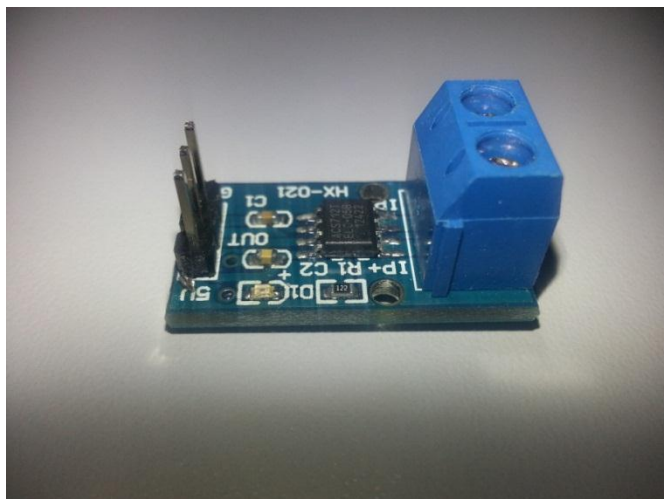


Figura 4.5 - Sensor de Corrente. Fonte: Autor

4.7 ROTEADOR WIELESS DI-524

Segundo site da do fabricante Dlink “O roteador Wireless DI-524 é capaz de transferir dados com uma taxa máxima de sinal wireless de até 150Mbps na frequência de 2.4GHz. O Roteador Wireless D-Link DI-524/150 também oferece quatro portas Ethernet para suportar múltiplos computadores que não possuem placa de rede wireless acoplada. A avançada tecnologia, incorporada ao Roteador Wireless DI-524/150, oferece velocidade de transferência de dados com um taxa de até 150Mbps. Seu firewall e suas criptografias são faceis de configurar e garantem total segurança aos usuários da rede. Recursos como filtragem de conteúdo, filtragem de MAC, bloqueio de URLs e bloqueio de domínios são ferramentas úteis para impedir que intrusos indesejáveis se conectem à sua rede ou naveguem por sites restritos.”

Neste projeto algumas funcionalidades do Ponto de acesso foram utilizadas, como a criptografia WPA 2 e o registro de endereços Mac.



Figura 4.6 - Ponto de Acesso. Fonte: Autor

4.8 APLICAÇÃO WEB

Como dito anteriormente, a proposta do projeto é controlar o protótipo de forma remota por uma aplicação que irá funcionar em qualquer dispositivo que tenha um navegador web, como por exemplo notebooks e celulares.

A aplicação foi feita de forma muito simples utilizando bibliotecas já prontas para os botões e uma lógica para controle. Abaixo temos uma foto da aplicação no celular.



Figura 4.7 - Aplicação. Fonte: Autor.

O primeiro botão – Abre Fechadura – irá mandar um pulso em nível alto para acionar o rele, que por sua vez irá acionar o fecho elétrico, depois de um certo tempo irá mandar um pulso em nível baixo para desligar o relé.

Os Três últimos botões – L/D Sala/Cozinha, L/D Quarto, L/D Banheiro – funcionam de forma muito similar, a diferença entra cada um deles é que controlam reles distintos. Ao clicar em um botão destes, o Arduino, através de sua programação verifica se a porta do comando está em nível alto ou baixo. Caso ela esteja em nível alto o pulso será de nível baixo e caso esteja em nível baixo o pulso será de nível alto. Desta forma ligando e desligando as lâmpadas dos cômodos do protótipo.

CAPÍTULO 5 – DESENHO E IMPLEMENTAÇÃO.

Este capítulo foi dividido em duas partes. Na primeira delas, Desenho do projeto, foi apresentado todos os critérios para planejar e desenhar o projeto, também foi colocado o porque de algumas tomadas de decisões para o projeto. Na segunda parte, Montagem e testes do projeto, foi apresentada a forma, em uma sequencia lógica, que tudo foi montado e testado. Este capítulo mostra bem os critérios e lógicas gerais utilizadas no projeto para desenho e implementação.

5.1 DESENHO DO PROJETO

Na figura 5.1 o projeto é apresentado uma boa parte do escopo, mostrando a infraestrutura e onde serão feitos alguns trabalhos, como, por exemplo, os cortes para implementação dos interruptores de três canais. Esta fase é de grande importância para que tudo seja colocado e montado em seus devidos lugares para que não seja encontrado surpresas na fase de implementação e, desta maneira, evitando retrabalho. Então tudo é pensado, visto de forma geral e desenhado.

Após todas as etapas da fase de desenho e planejamento, está completo todo o desenho na ferramenta de desenho técnico Autocad 2012, segue figura abaixo.

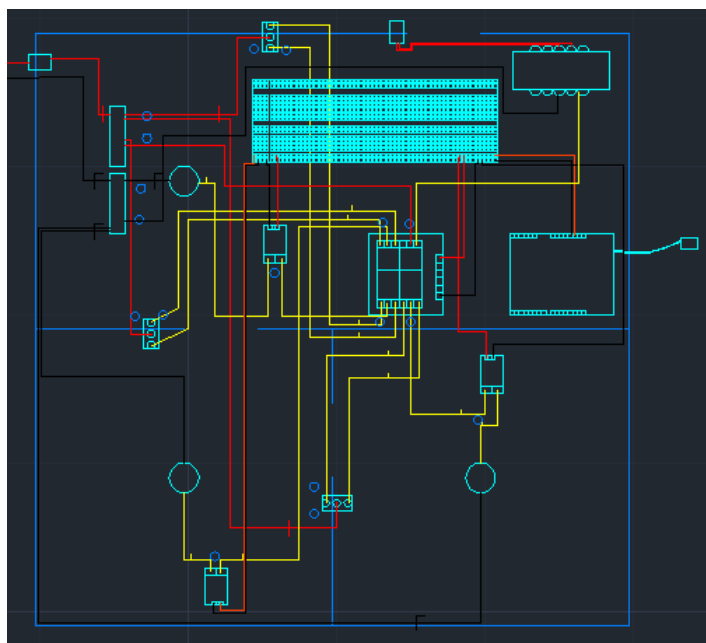


Figura 5.1- Desenho da planta no Autocad. Fonte: Autor.

5.2 PLANTA BAIXA E CORTES DO PROTÓTIPO

A ideia de um protótipo para espelhar um ambiente real é interessante, pois torna a experiência de apresentá-la e trabalhar nela muito mais compacta. Então, foi pensado em uma área para trabalhar de 40x40 centímetros. Desta forma o projeto se torna mais fácil de ser manipulado para lugares diferentes e, por seu tamanho, não precisa de materiais tão fortes e caros compondo sua estrutura.

O Material escolhido para criar a maquete do protótipo foi o papel panamá pelo seu baixo preço e sua resistência aos itens que ia suportar. Todo o desenho dos cortes foram feitos na ferramenta de desenho técnico Autocad. Segue abaixo imagem de todos os cortes feitos no papel panamá.

Também foi determinada a criação de um piso falso para que a visualização do projeto pronto ficasse melhor e interferências entre os circuitos de corrente contínua e alternada fossem reduzidas. Ficou então separado o circuito de corrente contínua sempre a cima do piso falso e o circuito de corrente alternada ficando abaixo do piso falso.

Além do piso falso para separar os circuitos elétricos, foi feito um primeiro andar para o Arduíno e o Ponto de Acesso, sempre com o objetivo de minimizar as interferências no projeto.

Foi decidido por fazer o corte das lâmpadas no piso da maquete para que ficasse mais simples a visualização de como tudo funciona. Ao ser colocado em um teto, dificultaria muito a manipulação e testes da maquete.

Cada corte foi identificado com uma letra e um número. Os buracos para interruptores, passagem de cabos e disjuntor termomagnético também foram desenhados.

5.3 DESENHO DO PROJETO ELETRÔNICO DO PROTÓTIPO

Para controle das lâmpadas e porta principal foram determinados locais específicos para cada componente de controle, interruptores e sensores de corrente. Foi decidido por colocar cada sensor de corrente e interruptor de luz em cômodos distintos do protótipo, digo isso, pois poderia ter sido colocado todos os sensores em um cômodo e passar os fios, mas desta forma pareceu mais organizado o desenho.

Para a fiação foram utilizados fios de cobre, conseguidos de cabos par trançado. Foi utilizado as cores marrom, verde e azul para Vcc, controle e ground respectivamente.

5.4 DESENHO DO PROJETO ELÉTRICO DO PROTÓTIPO

Na parte elétrica foi colocado um disjuntor termomagnético de 10A. Em um desenho anterior tinha sido colocado, antes do disjuntor termomagnético, um disjuntor diferencial residual em série, mas esse segundo foi retirado, pois no protótipo o quesito segurança não era tão importante quanto o orçamento.

5.5 ESCRITA DO CÓDIGO FONTE

Todo o código foi escrito na linguagem de programação do Arduino, que implementa Processing e C. De acordo com a pinagem utilizada definida na tabela de pinos utilizados é possível definir as diretivas de pré-compilação e as variáveis que serão utilizadas no decorrer do programa.

Também utilizei a biblioteca *EthernetSupW5100.h* para facilitar a criação dos botões da aplicação.

```
#include <SPI.h>
#include "EthernetSupW5100.h"

byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
IPAddress ip(192,168,1,150);
EthernetServer server(80);

int portaFechadura = 2;
int portaSalaCozinha = 6;
int portaQuarto = 8;
int portaBanheiro = 4;
int estado = 0;
```

Figura 5.2 - Parte do código fonte. Fonte: Autor

Após definir as bibliotecas e todas as variáveis que serão utilizadas no decorrer da programação, é preciso a parte do código que se repete várias vezes para que todo o controle funcione de forma correta. Na linguagem de programação do Arduino, a função setup é utilizada para o que se utiliza apenas na inicialização do sistema.


```

void setup()
{
  EthernetSupW5100.begin(mac, ip);
  server.begin();

  // Configurando portas dos botoes
  pinMode(portaFechadura, OUTPUT);
  pinMode(portaSalaCozinha, OUTPUT);
  pinMode(portaQuarto, OUTPUT);
  pinMode(portaBanheiro, OUTPUT);

  // Estado inicial das portas
  digitalWrite(portaFechadura, HIGH);
  digitalWrite(portaSalaCozinha, LOW);
  digitalWrite(portaQuarto, LOW);
  digitalWrite(portaBanheiro, LOW);

  // Registrando botoes
  //EthernetSupW5100.addButton(button pin, text on, text off, button type);
  EthernetSupW5100.addButton(portaFechadura, "Abre Fechadura", "", SWITCH_BUTTON);
  EthernetSupW5100.addButton(portaSalaCozinha, "L/D Sala/Cozinha", "", SWITCH_BUTTON);
  EthernetSupW5100.addButton(portaQuarto, "L/D Quarto", "", SWITCH_BUTTON);
  EthernetSupW5100.addButton(portaBanheiro, "L/D Banheiro", "", SWITCH_BUTTON);
}

```

Figura 5.3 - Parte do código fonte. Fonte: Autor

Essa função de inicialização basicamente define os pinos que serão utilizados para enviar o pulso elétrico, inicia o shield Ethernet para que o webserver fique escutando as requisições que serão feitas pelo navegador de internet de acordo com a necessidade do protótipo e por ultimo a utilização da função “EthernetSupW5100.addButton” da biblioteca EthernetSupW5100.h para criação dos botões na aplicação Web.

Esta função recebe 4 parâmetros. O primeiro deles é a porta em que o botão irá agir, configurada anteriormente como variáveis. O segundo parâmetro é o nome que será dado ao programa na aplicação Web. O terceiro não foi necessário para a proposta do projeto e, por isso não foi utilizado. O quarto botão é o tipo de botão que será utilizado, em nosso caso foi utilizado o botão do tipo “SWITCH_BUTTON”.

Depois destas inicializações, o sistema entra em laço de repetição infinito, definido na linguagem de programação Arduino com a função loop. A figura abaixo ilustra o laço de repetição.

```

void loop() {
    // Carrega HTML
    EthernetSupW5100.loadHtml(server);

    // Verifica se algum botao foi pressionado
    int lastButton = EthernetSupW5100.getLastClickedButton();
    byte state = EthernetSupW5100.getButtonState(lastButton);

    // Executa o comando conforme o botao clicado
    if (lastButton == portaSalaCozinha)
    {
        estado = digitalRead(portaSalaCozinha);
        if (estado == HIGH) {
            digitalWrite(portaSalaCozinha, LOW);
        } else {
            digitalWrite(portaSalaCozinha, HIGH);
        }
        //digitalWrite(portaSalaCozinha, state);
    }
    else if (lastButton == portaQuarto)
    {
        estado = digitalRead(portaQuarto);
        if (estado == HIGH) {
            digitalWrite(portaQuarto, LOW);
        } else {
            digitalWrite(portaQuarto, HIGH);
        }
    }
}

```

Figura 5.4 - Parte do código fonte. Fonte: Autor

Neste laço de repetição infinito tem-se o corpo principal do sistema, onde se irá tratar os comandos recebidos da aplicação web.

1. Carrega toda a página em HTML.
2. Verifica se algum botão da aplicação foi pressionado.
3. Executa os comandos conforme o dado recebido pela aplicação de controle.

5.6 MONTAGEM E TESTES DO PROJETO

A montagem e testes do projeto segue uma sequencia lógica para que tudo seja montado e funcione corretamente. As fases de montagem são importantes para tornar todo o processo mais ágio e organizado, de forma a não gerar problemas e retrabalho. Os testes após cada fase de montagem são importantes para a identificação preventiva e correção de erros em cada fase. Desta forma, caso um algum tipo de problema seja identificado, ele poderá ser resolvido imediatamente sem acumular problemas e confundir a implementação.



Figura 5.5 - Protótipo completo. Fonte: Autor

5.7 MONTAGEM E TESTES DA MAQUETE DO PROTÓTIPO

Toda a maquete do protótipo foi feita de um material, papel panamá. Este material foi cortado com um estilete e furado com uma furadeira, de acordo com todos os desenhos feitos no programa Autocad anteriormente. Logo após o corte, as partes foram montadas e coladas utilizando cola quente. Este processo de montagem foi realizado de baixo para cima.

5.8 MONTAGEM E TESTES DA PARTE ELÉTRICA E ELETRÔNICA DO PROTÓTIPO

Para suporte e prevenção contra curto circuito foi utilizado um disjuntor termomagnético de 10A. Todo o projeto elétrico está funcionando em uma rede com tensão de 110V estabilizada, foi ligado um estabilizador para diminuir a interferência do circuito de corrente alternada no circuito de corrente elétrica. Entre os interruptores de três terminais e os reles há uma ligação paralela, ligação tree way, para prover uma independência do sistema, caso o controle de forma remota venha a falhar por algum motivo, o as luzes podem ser controladas de forma independente.

5.9 MONTAGEM E CONFIGURAÇÃO DA TOPOLOGIA DA REDE

Foram utilizados duas funcionalidades no que diz respeito a segurança do acesso ao sistema para a comunicação entre os dispositivos que possuíam a aplicação e o ponto de acesso. Isso é de grande importância para que qualquer pessoa não acesse o sistema pela internet ou até mesmo pela própria wireless e faça algo que vá afetar a integridade do sistema.

Primeiramente foi habilitado a conexão via rede wireless com o ponto de acesso, configurado a criptografia WPA2 e definido sua chave de segurança como segue na imagem.

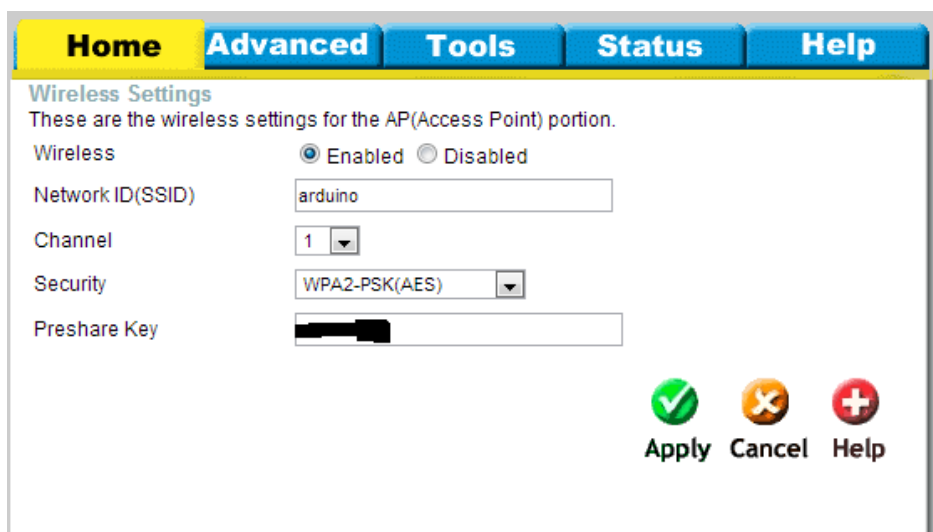


Figura 5.6 - Configuração da criptografia utilizada. Fonte: Autor

A segunda configuração foi o registro do endereço MAC no ponto de acesso para que seja criado um filtro de acesso e, desta forma, apenas os equipamentos com os endereços pré-configurados tivessem acesso ao sistema.

Home **Advanced** **Tools** **Status** **Help**

Filter
Filters are used to allow or deny LAN users from accessing the Internet.

☐ IP Filters ☐ URL Blocking
☒ **MAC Filters** ☐ Domain Blocking

MAC Filters
Use MAC address to allow or deny computers access to the network.

☐ Disabled MAC Filters
☒ Only **allow** computers with MAC address listed below to access the network
☐ Only **deny** computers with MAC address listed below to access the network

Name

MAC Address - - - -

DHCP Client -- select one --




  
Apply **Cancel** **Help**

Figura 5.7 - Configuração do filtro de endereço MAC. Fonte: Autor

CAPÍTULO 6 – RESULTADOS OBTIDOS

Neste capítulo é apresentado algumas simulações com o projeto seguindo a proposta que foi apresentada. Também serão apresentados neste capítulo alguns problemas encontrados no sistema durante a realização de todo o trabalho.

6.1 TESTES COM O SISTEMA

Nessa seção, são realizadas algumas simulações e testes com todas as funcionalidades propostas, no intuito de alcançar o mesmo comportamento de uma situação em ambiente real. As funcionalidades testadas tiveram como objetivo alcançar as seguintes metas:

- Testes de toda parte elétrica, por questão de segurança de manuseio e dos componentes eletrônicos;
- Simulações para ligar e desligar as luzes através do controle por Smart Fone e notebook com navegadores diferentes;
- Simulações para fazer o acionamento da fechadura através do controle por Smart phone e notebook com navegadores diferentes;

6.2 TESTES COM A PARTE ELÉTRICA

Antes de realizar as simulações do shield Ethernet em funcionamento com o Arduino e módulo reles. Toda a parte elétrica foi testada por partes antes, seguindo recomendações da orientadora. Abaixo a figura 6.1 ilustra o teste realizado em um dos circuitos para verificação da tensão em 110V.

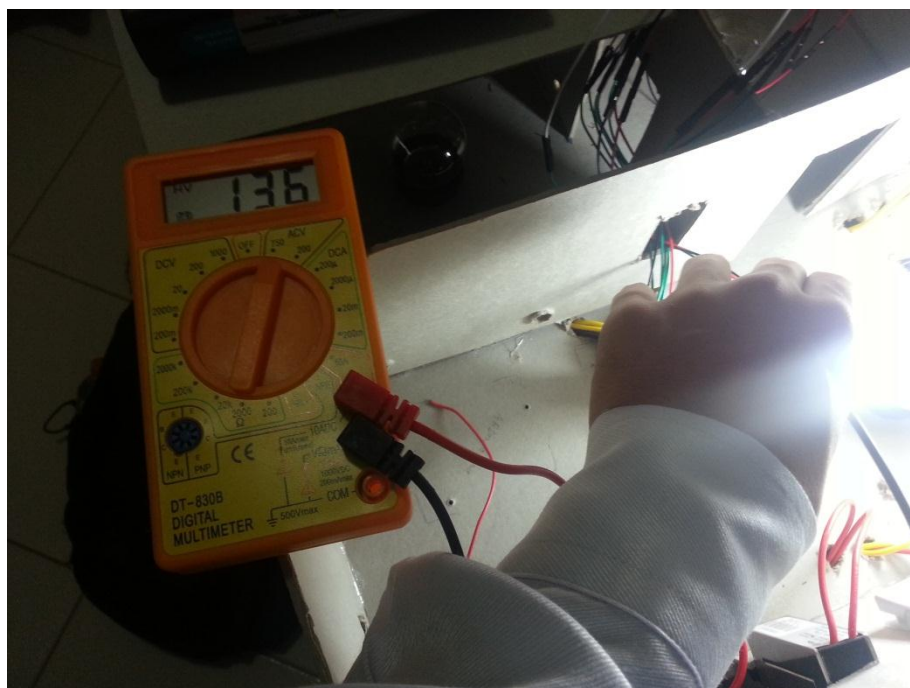


Figura 6.1 - Teste de circuito com multímetro. Fonte: Autor

Este procedimento foi realizado para o circuito de cada lâmpada. Começando pelo circuito da Sala/Cozinha, quarto, banheiro e fecho elétrico. Por ter seguido exatamente as orientações do professor, não foram encontrados problema algum, como fuga de corrente ou curto circuito, na montagem da parte elétrica.

6.3 TESTES COM O CONTROLE

Após ter testado toda a parte elétrica dos circuitos garantindo a segurança dos componentes, foi feita a verificação do funcionamento dos componentes eletrônicos. A figura 6.2 mostra os componentes em funcionamento.

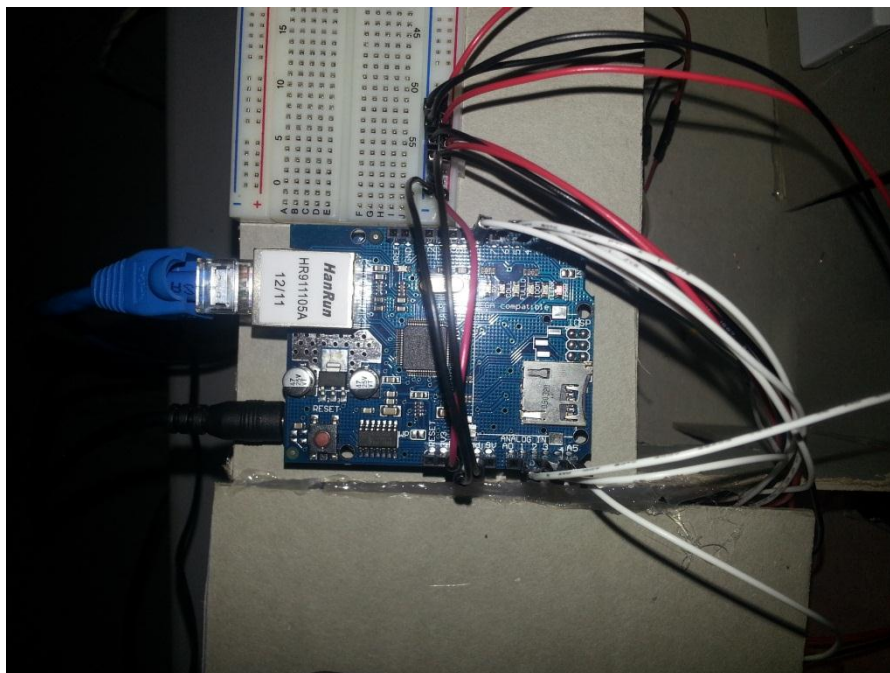


Figura 6.2 - Shield Ethernet em funcionamento com Arduino Uno. Fonte: Autor

Todo o acionamento dos 4 reles foram testados antes da parte elétrica. O barulho que o equipamento faz ao acionar a bobina mostra que está tudo funcionando conforme o esperado.

6.4 ACIONAMENTO DAS LUZES E PORTA PRINCIPAL

Esta simulação tem como finalidade o funcionamento do sistema desde o momento em que é ligado, acessado por um dispositivo controlador e feito o controle do sistema como o proposto pelo trabalho. Todo esse procedimento foi realizado com um notebook e um Smart phone. O autor, que realiza os comandos através da interface do navegador segue as seguintes etapas.

1. Ligar o produto utilizando uma bateria de 9V.

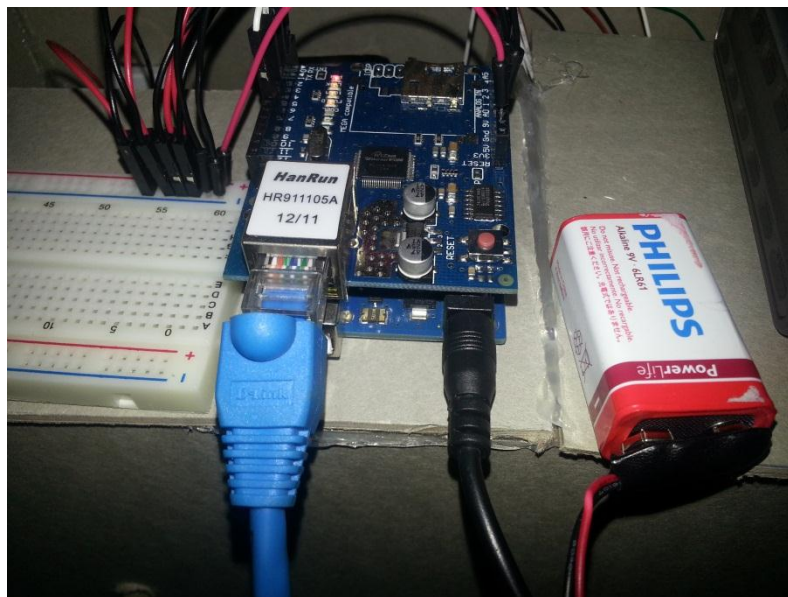


Figura 6.3 - Alimentação de 9v para o Arduino. Fonte: Autor

2. Acessar a rede através da Wi-fi com o SSID “arduino” e o sistema através dos navegadores de internet. Este procedimento foi feito com os navegadores Firefox e Google Chrome.

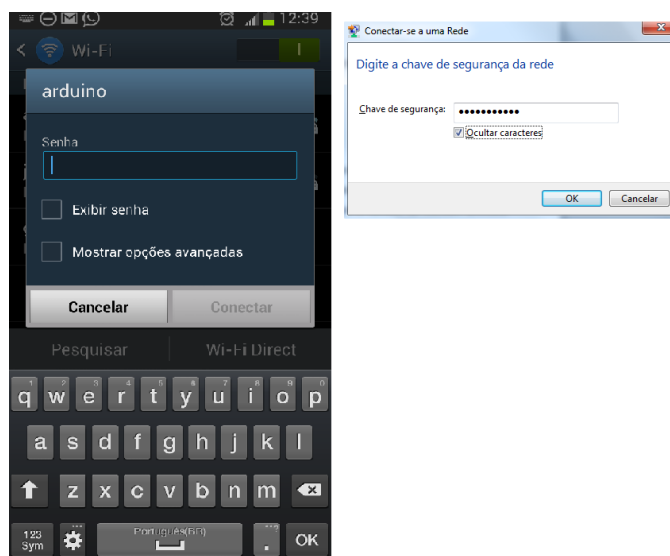


Figura 6.3 - Senha de segurança para acesso nos diferentes dispositivos. Fonte: Autor.

3. Clicar nos comandos, tanto no notebook quanto no smart phone, para verificar o controle de forma remota do sistema, porta principal e lâmpadas.



Figura 6.4 - Painel de controle em ambos os equipamentos. Fonte: Autor.

4. Fazer o acionamento manual diretamente no interruptor para verificar se as lâmpadas funcionam de forma independente do controle remoto.

6.5 PROBLEMAS ENCONTRADOS

Durante o desenvolvimento do sistema, alguns problemas foram aparecerem ao que se refere a interferências internas no protótipo. Inicialmente foi verificado que um protótipo diretamente ligado a uma tensão de 220V fazia com que o circuito de corrente alternada gerasse uma interferência muito grande no circuito de corrente contínua do protótipo, principalmente por, inicialmente, o transformador utilizado para transformar a corrente alternada de 220V em corrente alternada de 12V para alimentar o fecho elétrico estar muito perto do Arduino. Isso fez com que o módulo de 4 reles alternasse de seu estado normalmente fechado para normalmente aberto sem o acionamento do celular ou notebook. Este problema foi sanado aplicando-se 3 medidas simples. A primeira delas foi modificar o local onde a plataforma microcontrolada ficava, levando-a para um andar a cima do protótipo. Logo após

foi diminuir a tensão de 220V para 110V e, por fim, estabilizar todo o circuito de corrente alternada do protótipo ligando-o em um estabilizador.

Outro problema encontrado é referente aos sensores de corrente utilizados nos retornos das 3 lâmpadas do protótipo e ligados as portas analógicas do Arduino. Estes componentes foram ligados de forma correta, mas não enviaram valores distintos dos normais quando tinha corrente ou não nos fios. Este problema seria facilmente solucionado substituindo os sensores por sensores mais precisos, mas infelizmente esta medida iria aumentar bastante o valor do projeto e não foi adotada. Mesmo com a solução diminuindo muito as interferências sob os fios, diminuindo a tensão para 110V, desta forma aumentando a corrente nos fios, os sensores não pegaram os valores esperados para serem tratados pelo Arduino. Mesmo assim, os sensores foram mantidos apenas para mostrar o local de seu funcionamento e que iram servir para trabalhos futuros.

Por fim, outro problema encontrado por muitos desenvolvedores de sistemas do mesmo tipo é a criação de todos os algoritmos e bibliotecas de seu início. Na comunidade que contribui para o desenvolvimento de aplicações para esta plataforma existem muitas bibliotecas já prontas e que foram reaproveitadas para serem utilizadas neste trabalho.

CAPÍTULO 7 – CONSIDERAÇÕES FINAIS

7.1 CONCLUSÕES

Neste projeto foi desenvolvido um protótipo com o objetivo de mostrar como a automação torna mais ágil alguns processos e procedimentos trabalhando com o controle de luminárias e porta principal através de uma plataforma microcontrolada controlada por qualquer dispositivo que tenha um navegador web.

Mesmo com os problemas encontrados na implementação deste projeto, a ideia proposta foi alcançada. Através de planejamento e desenho de todo o protótipo foi evitado retrabalho, um dimensionamento mais eficiente dos componentes necessários e uma agilidade maior na montagem do projeto.

Foram utilizadas muitas matérias do curso de Engenharia De Computação para desenvolvimento e implementação do projeto. Desta forma podemos observar o quanto é importante a visão das matérias sendo aplicadas em conjunto e não de forma independente para alcançar determinados objetivos.

Os resultados obtidos com o desenvolvimento do protótipo torna possível concluir que é possível chegar até a proposta apresentada no início deste trabalho e que o projeto está funcionando de acordo com o planejado e programado.

7.2 PROPOSTAS PARA TRABALHOS FUTUROS

Automação residencial está ganhando sua parte no mercado com os mais diversos dispositivos e equipamentos. Com o avanço das tecnologias utilizadas em automação Predial temos, cada vez mais, produtos mais integrados e com preços mais acessíveis, desta forma proporcionando uma atualização dos projetos recentes. Algumas propostas de desenvolvimento para este projeto são:

- Verificação dos estados das luminárias na própria aplicação web. Com um painel informando o estado em tempo real das luminárias tornaria mais intuitiva a utilização do sistema de controle.
- Verificação dos estados da porta principal. Através de sensores verificar os seguintes estados da porta principal: Aberta, fechada, trancada e destrancada.

- Implementar um sistema com baterias. Para que, caso caia a energia, o sistema continue em pleno funcionamento, desta forma aumentando sua disponibilidade.

REFERÊNCIAS BIBLIOGRÁFICAS

TANENBAUM, Andrew S. Redes de computadores.

TORRES, Gabriel. Redes de Computadores

LEMAY, Laura; COLBURN, Rafe; TYLER, Denise. Aprenda a Criar Páginas Web com HTML e XHTML em 21 dias.

GIMENEZ, Salvador. Microcontroladores 8051.

MCROBERTS, Michael. Arduino Básico.

ARDUINO, Site do Projeto Arduino. Arduino. Disponível em: <arduino.cc/> Acesso: 8 out. 2013.

SENSOR ACS712, Site do projeto. Disponível em: <<http://pdf1.alldatasheet.com/datasheet-pdf/view/168326/ALLEGRO/ACS712.html>> Acesso: 8 out. 2013.

SHIELD ETHERNET, Site do Projeto. Disponível em: <<http://arduino.cc/en/Guide/ArduinoEthernetShield>> Acesso: 10 out. 2013.

BIBLIOTECAS SERIAL LINK, Site da empresa. Disponível em: <www.seriallink.com.br> Acesso: 10 out. 2013.

DLINK, Site do fabricante. Disponível em: < <http://www.dlink.com.br/produtos-detahes/items/di-524.html>> Acesso: 20 out. 2013.

ELETRICAINFO, site. Disponível em: < <http://www.eletrica.info/three-way-interruptor-paralelo-o-que-e-como-funciona/>> Acesso 20 out. 2013.

APÊNDICE A – GASTOS DO PROJETO

Componente	Valor Unitário	Preço	Total
Transformador 220/110V para 12V 1A	1	R\$ 23,00	R\$ 23,00
Fechadura elétrica	1	R\$ 45,50	R\$ 45,50
Arduino Uno	1	R\$ 60,00	R\$ 60,00
Shield Ethernet	1	R\$ 59,90	R\$ 59,90
Bateria de 9 v	3	R\$ 2,50	R\$ 7,50
Módulo rele 4 canais	1	R\$ 34,90	R\$ 34,90
Sensor de corrente	3	R\$ 24,50	R\$ 73,50
Papel panamá	3	R\$ 6,00	R\$ 18,00
Lâmpada bol e-27 15w cl sdk/empalux	3	R\$ 3,75	R\$ 11,25
Soquete e-14 cerâmico c/ rabinho sdk	3	R\$ 5,83	R\$ 17,49
Il int paralelo 10A 5ta9 9001	3	R\$ 7,98	R\$ 23,94
Il placa 4x2 3 mod br 5ta9 9026 ilus	1	R\$ 3,74	R\$ 3,74
Cabinho flexível 750v 02,5mm mt branco	2	R\$ 1,25	R\$ 2,50
Cabinho flexível 750v 02,5mm mt preto	2	R\$ 1,25	R\$ 2,50
Cabinho flexível 750v 02,5mm mt vermelho	2	R\$ 1,25	R\$ 2,50
Disjuntor mono 010A AMER	1	R\$ 9,10	R\$ 9,10
Estilete	1	R\$ 7,00	R\$ 7,00
Pistola de cola	1	R\$ 10,90	R\$ 10,90
Multímetro digital dt-830B	1	R\$ 17,00	R\$ 17,00
AP - D-Link DI-524	1	R\$ 63,00	R\$ 63,00
Total:			R\$ 493,22

APÊNDICE B – CÓDIGO FONTE

```
#include <SPI.h>
#include "EthernetSupW5100.h"

byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
IPAddress ip(192,168,1,177);
EthernetServer server(80);

int portaFechadura = 2;
int portaSalaCozinha = 6;
int portaQuarto = 8;
int portaBanheiro = 4;
int estado = 0;

void setup()
{
  EthernetSupW5100.begin(mac, ip);
  server.begin();

  // Configurando portas dos botoes
  pinMode(portaFechadura, OUTPUT);
  pinMode(portaSalaCozinha, OUTPUT);
  pinMode(portaQuarto, OUTPUT);
  pinMode(portaBanheiro, OUTPUT);

  // Estado inicial das portas
  digitalWrite(portaFechadura, HIGH);
  digitalWrite(portaSalaCozinha, LOW);
  digitalWrite(portaQuarto, LOW);
  digitalWrite(portaBanheiro, LOW);

  // Registrando botoes
  //EthernetSupW5100.addButton(button pin, text on, text off, button type);
  EthernetSupW5100.addButton(portaFechadura, "Abre Fechadura", "", SWITCH_BUTTON);
  EthernetSupW5100.addButton(portaSalaCozinha, "L/D Sala/Cozinha", "", SWITCH_BUTTON);
  EthernetSupW5100.addButton(portaQuarto, "L/D Quarto", "", SWITCH_BUTTON);
  EthernetSupW5100.addButton(portaBanheiro, "L/D Banheiro", "", SWITCH_BUTTON);
}

void loop()
{
  // Carrega HTML
  EthernetSupW5100.loadHtml(server);

  // Verifica se algum botao foi pressionado
  int lastButton = EthernetSupW5100.getLastClickedButton();
```



```

byte state = EthernetSupW5100.getButtonState(lastButton);

// Executa o comando conforme o botao clicado
if (lastButton == portaSalaCozinha)
{
    estado = digitalRead(portaSalaCozinha);
    if (estado == HIGH) {
        digitalWrite(portaSalaCozinha, LOW);
    }else {
        digitalWrite(portaSalaCozinha, HIGH);
    }
    //digitalWrite(portaSalaCozinha, state);
}
else if (lastButton == portaQuarto)
{
    estado = digitalRead(portaQuarto);
    if (estado == HIGH) {
        digitalWrite(portaQuarto, LOW);
    }else {
        digitalWrite(portaQuarto, HIGH);
    }
    //digitalWrite(portaQuarto, state);
}
else if (lastButton == portaBanheiro)
{
    estado = digitalRead(portaBanheiro);
    if (estado == HIGH) {
        digitalWrite(portaBanheiro, LOW);
    }else {
        digitalWrite(portaBanheiro, HIGH);
    }
    //digitalWrite(portaBanheiro, state);
}
else if (lastButton == portaFechadura)
{
    digitalWrite(portaFechadura, LOW);
    delay(1000);
    digitalWrite(portaFechadura, HIGH);
}
}

```

APÊNDICE C – ESQUEMÁTICO: ARDUINO UNO

